Electronic Version

Stylesheet Version v1.1.1

Description

[System and Method for Providing Secure Testing Aids]

BACKGROUND OF INVENTION

[0001] In its best mode present invention is directed to the use of a handheld computer containing a flight calculator as an approved device for airman testing in accordance with Order 8080.6C of the FAA (Federal Aviation Administration).

[0002] This Order defines the devices which can be used as testing aids during the airman testing, and their characteristics. A number of hard-wired, prior-art calculators have been approved for use with airman testing, and it is desired to implement a program which would be acceptable to the FAA for use as a testing aid when resident on a handheld computer.

[0003] Because of the flexibility of a handheld computer, however, there is a danger that a program having approved characteristics, and resident on the computer may be excluded from approval, because the handheld computer

possesses other capabilities which can be used to circumvent testing security. For instance, a handheld computer may be used by an unscrupulous testee to illicitly store test answers in violation of the FAA rules.

[0004] The present invention provides a method to allow the use of a handheld computer for testing purposes under the FAA rules by including a verification of the handheld computer contents at the time of testing. This verification insures that:

[0005] (1)the ability of the handheld computer to communicate or network is disabled;

[0006] (2)the handheld computer contains only those programs previously approved for testing; and

[0007] (3)the handheld computer"s RAM, ROM and external memory are placed in a known, approved state.

[0008] After verification has been passed, the present invention provides for restoring the handheld computer to its original condition for general use.

[0009] Although the present invention was inspired by the requirements of FAA airman testing, it clearly has uses in other applications beyond the FAA testing requirements.

[0010]

SUMMARY OF INVENTION

[0011] It is an object of the present invention to provide a method for testing, using a handheld computer, wherein the pre-existing programs of the handheld computer are exchanged for a new set of programs configured specifically for said testing. It is a further object of this invention to provide such a method which contains a verification procedure which assures a proctor supervising the test that the program contents of the handheld computer contain only programs approved for use in the test.

[0012] In accordance with a first aspect of the invention, the method for testing utilizes a handheld computer including an application launcher and a multiplicity of previous applications.

[0013] In accordance with a second aspect of the invention the method includes removing those applications that are unrelated to airman testing, loading one or more new applications into the handheld computer, loading a verification application into the handheld computer, launching, by the proctor, of the verification application and verifying whether or not the handheld computer contains only an approved set of applications and the launcher, allowing the testee to proceed with the test if and only if it does contain only the approved set of applications and

launcher.

[0014] In accordance with a third aspect of the invention, after proceeding to take the test, the testee erases the applications used for testing from the handheld computer and restores the previous applications to the handheld computer.

[0015] In accordance with a fourth aspect of the invention, the method launcher and a multiplicity of previous applications are all originally stored in an internal memory in the form of an old image, and an external memory containing a new image is connected to the handheld computer, the new image containing a verification application and one or more new applications, and the new image on the external memory is exchanged with the old image on the internal memory.

[0016] In accordance with a fifth aspect of the invention, the testee exchanges the old image on the external memory with the new image on the internal memory after testing, thereby restoring the handheld computer to its original configuration.

[0017] In accordance with a sixth aspect of the invention the verifying further includes the steps of performing a checksum calculation on one or more portions of the internal

memory, and comparing the resulting checksum to a desired checksum.

[0018] In accordance with a seventh aspect of the invention the verifying further comprises comparing a desired key code with a key code generated by the verifying step.

[0019] In accordance with an eighth aspect of the invention the verifying includes confirming that no network devices of the handheld computer are enabled, and that no communications devices of the handheld computer are enabled.

[0020] In accordance with a ninth aspect of the invention the value of the generated key code is a dynamic value.

[0021] In accordance with a tenth aspect of the invention the dynamic value is generated by a pseudo-random number generator, comprising a PRNG algorithm, in the handheld computer, and is compared to a value generated by the same pseudo-random number generator, comprising the PRNG algorithm, outside of the handheld computer.

[0022] In accordance with an eleventh aspect of the invention the internal memory of the handheld computer holding the applications is a flash ROM memory or any other semi-permanent memory type.

[0023] In accordance with a twelfth aspect of the invention the external memory of the handheld computer can be a re-

movable memory module.

[0024] In accordance with a thirteenth aspect of the invention the verification code is different for each of a number of different test centers.

[0025] In accordance with a fourteenth aspect of the invention the verifying is performed by a verification application which incorporates encryption to some or all of the verification application.

[0026] In accordance with a fifteenth aspect of the invention the new applications include a flight calculator.

Brief Description of Drawings

[0027] These, and further features of the invention, may be better understood with reference to the accompanying specification and drawings depicting the preferred embodiment, in which:

[0028] Figure 1 depicts the hardware configuration required for the download embodiment of the invention.

[0029] Figure 2 depicts the configuration of the typical software contents of a typical handheld computer.

[0030] Figure 3 depicts a flow chart of the download embodiment of the invention.

[0031] Figure 4 depicts a flow chart of the verification execution of the embodiment using a handheld computer having a

flash-ROM internal memory and an external memory module.

[0032] Figure 5 depicts a flow chart of the complete operation of the embodiment using a handheld computer having a flash-ROM internal memory and an external memory module.

DETAILED DESCRIPTION

[0033] The present invention in its best mode is a method for using a prior-art handheld computer as a host for flight computer software which can be used in conjunction with FAA certified testing. In more general embodiments, the method may be used in conjunction with any prior-art computer running a testing aid application which can be used in conjunction with any certified testing process.

[0034] In the subsequent description the term handheld computer will be used interchangeably with the term palm-top, and used to mean any general purpose hand held computer or device having computational capabilities.

[0035] The invention further provides a secure program, contained within the palm-top, which verifies to a test proctor that the palm-top does not contain any means for the user to view previously stored test answers in violation of the FAA rules. And finally the invention provides for a

means to store pre-existing palm-top applications on a remote computer, to erase those pre-existing applications from the palm-top, to download the flight calculator program from the remote computer, and to restore the pre-existing program to the palm-top after the testing.

[0036] The invention further provides a secure program environment contained in the handheld computer which verifies to a test proctor that the handheld computer does not permit the user to access any previously stored data or store any data during the exam. And finally the invention provides for a means to install the secure program environment and then restore the handheld computer to its original state after testing.

[0037] The handheld computer typically contains a number of software components stored in an internal memory of the handheld computer which is typically erasable. A popular common implementation provides an erasable flash memory, called herein the internal flash ROM, which contains these software components, and maintains them even when power to the handheld computer is removed. The handheld computer is usually protected against inadvertent erasure of the internal flash ROM by making such erasure very difficult for a non-sophisticated user. How-

ever, this internal flash ROM can be erased, and reloaded with other software which may execute properly on the handheld computer, providing that such software is properly designed. The current invention provides for the erasing of the software resident on the internal flash ROM and its replacement by other software which contains the functions required for this invention.

[0038] An alternative to the erasing of the internal flash ROM and subsequent reloading of new data into the internal flash ROM can be accomplished in one step by simply overwriting the image contained on the internal flash ROM with a complete new internal flash ROM image.

[0039] In addition to the internal flash ROM which contains the handheld computer software, the typical handheld computer also contains RAM which can be read or written by the handheld computer software, but which is volatile, so that it is erased when the handheld computer is switched off.

[0040] Figure 2 shows the software components of a typical handheld computer. The launcher 10 is a type of operating system which controls input from the user through the handheld computer's keyboard, touch-screen, infra-red, memory, and input port devices. In addition, the launcher

also provides means to load one of the applications programs into a section of RAM memory, and to begin execution of the loaded application. When an application is loaded, previously loaded applications are usually overwritten.

[0041] An alternative form of launcher executes the program applications directly from their locations in the internal flash ROM.

[0042] All of the software contained within the handheld computer, whether part of the operating system (launcher) or application software, is stored in the internal flash ROM of the handheld computer. In addition, most handheld computer"s have the ability to incorporate other applications via additional plug-in memory modules such as secure digital, or SD cards, which can be incorporated without disassembling the handheld computer.

[0043] A typical set of applications are shown in Figure 2. These include a simple calculator 12, time and date calculator 14, time-zone calculator 14, address book 16, appointment calender 18, Email processor 20, external synchronization processor 22, Memo Pad 24, and WIFI communications Module 28.

[0044] Several of these applications have the ability to input and

store data, and to display the data subsequently. The memo pad, for instance, contains a primitive word processor, and the email processor allows the user to input data to be emailed out, and to display data which has been emailed in, presumably over a wireless or WiFI®connection. The data from these applications is generally stored in the internal flash ROM, so that it is not lost when the handheld computer is powered down.

[0045] In addition most handheld computers provide for the downloading and subsequent execution of additional applications not originally installed on the handheld computer. In the instant case, one or more applications are provided in the handheld computer to be used by pilots to perform flight planning, navigation and other performance computations necessary for safe flight. The handheld computer and the applications will be referred to hereinafter collectively as the flight calculator.The flight calculator is designed to be used as an approved device for airman testing in accordance with Order 8080.6C of the Federal Aviation Administration, entitled "Conduct of Airman Knowledge Tests". This order provides, *inter alia*, that "Testing centers may provide calculators to applicants and/or deny applicants" use of their personal calculators

based on the following limitations:(1) Prior to, and upon completion of the test, while in the presence of the proctor, the applicant must actuate the ON/OFF switch and perform any other function that ensures erasure of any data stored in memory circuits.

[0046] (2)The use of electronic calculators incorporating permanent or continuous type memory circuits without erasure capability is prohibited. The proctor may refuse the use of the applicant"s calculator when unable to determine the calculator"s erasure capability."The purpose of these regulations is to prevent the test-taker from using a device which could contain the test answers in other portions of the calculators which have the ability to input and store text, and thus circumvent the purpose of the test.

[0047] A handheld computer-based flight calculator would ordinarily not be able conform to these requirements, since, as indicated above, commonly-used handheld computers have the ability to input and store data, and further do not possess the ability to clear the data stored in selected applications with the stroke of a single button.

[0048] The current invention, however, provides a means adapting a handheld computer to the requirements of Order 8080.6.

[0049] *Download Embodiment*

[0050] The embodiment of the invention may be understood by first referring to Figure 1, which depicts a personal computer 2 attached to a palm-top handheld computer (personal digital assistant) 4 by means of a cradle 6, which is, in turn, connected to the personal computer via a USB cable 8. The cradle allows a simplified physical interface for the handheld computer, and typically contains a switch 8 which commands the synchronization of data between the personal computer and the handheld computer. The handheld computer is electrically connected to the cradle via a comb of external contacts 7 located on the bottom end of the handheld computer, which mates with a connector 9 mounted on the cradle.

[0051] The method may be understood in one of its embodiments by referring to the flow chart of Figure 3.

[0052] This figure assumes that the PC Host software which controls the process has first been loaded into the computer 2, and is running.

[0053] The first step, as seen in Figure 3, is to upload 20 the handheld computer applications from the handheld computer and store them in the computer, where they can later restored to the handheld computer. Next, the hand-

held computer applications 22 are erased from the handheld computer, and the RAM is likewise cleared. This erasure may be total, or partial, wherein only those applications with the capability of storing and displaying text information are erased. Further, the erasure may be commanded by the computer, or, alternatively, a small application may be downloaded into the handheld computer which, in turn may be activated to erase the applications. Note further that this erasure may be done in advance of the testing, or may be commanded at the test center at the time of the test.

[0054] The flight calculator is next downloaded 24 into the handheld computer, together with a verification program. The verification program may be included as part of the flight calculator, or it may be a separate program module, which has the advantage that the flight calculator, when loaded for use other than testing, takes up less memory in the handheld computer when the verification program is not included. At this point, the handheld computer is in condition to be used for FAA airman testing.

[0055] The next steps provide verification at the test site that the handheld computer is in condition for testing. This verification has the purpose of insuring that the flight calcula-

tor is the program approved by the FAA, and further that the handheld computer does not contain any programs capable of storing and displaying the test answers, many of which are available in advance to the persons being tested (the "testees").

[0056] The proctor starts the verification process 26 by executing a checksum calculation of the internal flash ROM image of the handheld computer. The checksum also includes the handheld computer RAM, which is cleared at the time that the internal flash ROM image was erased, or swapped for a new image.

[0057] If the checksum test produces the correct checksum, which is stored within the verification program, then the verification program produces a verification or key code, which the proctor checks against the code supplied to him by the FAA (the FAA code). If the checksum test is passed, and the key code is correct, then the proctor allows the testee to take the test 36. Otherwise, the handheld computer is judged to be illegal for testing purposes, and the testee may proceed to take the test, but without use of the handheld computer-based flight calculator.

[0058] At the end of the test the proctor again is given the handheld computer containing the flight calculator, and exe-

cutes an erasure of the RAM memory 38 before allowing the testee to depart the test center.

[0059] Finally, after the last step has been completed, the testee may restore the handheld computer to its pre-test configuration by again connecting the erasing the applications of the internal flash ROM and downloading an image of the original internal flash ROM onto the internal flash ROM 40.

[0060] *Removable Memory module Image Embodiment*

[0061] As an alternative to downloading the flight calculator and verification applications from a computer onto the handheld computer internal flash ROM which contains all of the applications which reside on the handheld computer, in addition to the launcher.

[0062] In this embodiment the flight calculator and verification program are not downloaded by the testee by the link shown in Figure 1, but are contained on an external removable memory module inserted into the handheld computer in an externally accessible port designed to allow additional applications to be added to the handheld computer without having to download them into the internal flash ROM internal to the handheld computer.

[0063] The use of such removable memory modules is well

known in the prior art and need not be described here in further detail.

[0064] In this embodiment the external memory module contains an application which permits a swapping of the image of the internal flash ROM with the flight calculator-verification image contained on the external memory module.

[0065] The operation of this embodiment may be understood by referring now to Figure 5. The flow chart of this Figure begins when the testee, prior to arriving at the test site, first inserts the Configuration external memory module into the external port of the handheld computer 21. This Configuration card contains an application which performs the swapping 42 the image of the internal flash ROM with the flight calculator-Verification image contained on the Configuration removable memory module.

[0066] Next, the Configuration application is executed 23, which actually performs the swapping of the two images.

[0067] Once the swapping is performed, the method proceeds as in the previous embodiment, in which the reference number of the steps and the descriptions within the flow chart elements are the same as in said previous embodiment, beginning with the Checksum Calculation Module (reference number 26).

[0068] In order to perform the verification that the key code corresponds to the FAA code, a number of options are available, which provide an increasing level of security, and are described as follows.

[0069] *Primary Verification Embodiment*

[0070] After the software of the present invention has been swapped with the software previously residing on the handheld computer, the verification procedure may begin.

[0071] The software of the handheld computer is launched when the handheld computer is powered up. Upon launch, the software of the present invention will perform the functions shown in the flow chart of Figure 4.

[0072] A system cold boot or hard reset 60 is first performed. This cold boot will clear all volatile memory (RAM) on the handheld computer device 62.

[0073] Next, a checksum of both the internal flash ROM and all RAM is performed 64 in order to verify that the ROM and RAM are in a known, approved state before proceeding, and the checksum calculated is compared to a value stored in the flash-ROM memory.

[0074] The system next checks that no external memory media, such as external SD cards, are currently inserted 66.

[0075] Finally, the system checks to make sure that any network

devices and external communication devices are disabled 68. The software of the present invention, which is now the only software resident on the handheld computer, does not have any capability to enable any of the communication or networking facilities of the handheld computer.

[0076] The verification program next tests to determine if any of these tests fails 70, and if so the program will not continue, but indicate a failure. Otherwise, the verification program generates a code, and prompts the test administrator, or proctor for acceptance. The proctor checks this code against a code obtained from the FAA 74. If the codes correspond, then the proctor will allow the testee to proceed with the test, but otherwise the testee will not be allowed to use the flight calculator to used for the test.

[0077] In an alternative version of this embodiment code generated by the handheld computer, and that supplied by the FAA will be a dynamic value which changes in accordance with a formula dependent upon one or more variables supplied to the handheld computer verification program. For instance, a pseudo-random number generator may be used which uses a variable input supplied by the FAA, together with the FAA code, prior to verification. The verifi-

cation of step 74 requires the proctor to input the variable, after which the code is generated using the same pseudo-random number generator algorithm as that used by the FAA in generating the FAA code.

[0078] *First Verification Input Embodiment*In this embodiment, the proctor supervising the testing starts the verification application which appears on the screen of the handheld computer. The verification application begins, and requests a verification key, which is a code, or password, known only to the proctors. In this embodiment, a single verification key is used which is effective for all testing centers. When the proctor keys in the proper verification key, the verification application begins a verification check of all of the applications still residing on the handheld computer. Typically this verification check is a type of checksum check of all of these remaining applications. If the checksum check produces a result which equals a value stored in the verification program, then the application produces a message on the handheld computer stating that the verification has succeeded. Otherwise, the handheld computer produces a "FAILURE" message. Other output message embodiments are discussed *infra*.

[0079] The alternative to this procedure is to perform the check-

sum calculations first, and if the checksum is correct, to check the verification key before allowing the testing to proceed.

[0080] The use of this verification key insures that a testee may not run the verification application, but that only a proctor may do so. Preventing the testee from running the verification application makes it difficult for the testee to understand the operation of the verification application, and thus circumvent the verification features.

[0081] *Second Verification Input Embodiment*

[0082] In a further embodiment, there is a verification key corresponding to each testing center, and the proctor at each such center is apprised of the corresponding verification key. In addition to the verification key for a specific testing center, the name of the center, or its zip code, or other commonly used location code is also keyed in to the verification program. Thus, the proctor must not only have the verification key, but the verification key which corresponds to his or her particular testing center, in order to allow the checksum testing to proceed, and eventually produce a "SUCCESS" message on the handheld computer. Thus a table of all of the verification keys must be contained within the verification program.

[0083] *Third Verification Input Embodiment*In a still further embodiment a pseudo-random code generator (PRCG) is utilized to generate a verification key in real time, so that the verification code for each test site need not be input in advance to the verification program. The use of a verification key which is not a fixed, preset value in the handheld computer is referred to as a dynamic key.

[0084] This embodiment requires the use of a program running on a local computer at the testing center, or somewhere within the testing organization, which produces the verification key and makes the verification key accessible to the proctor. The PRCG will generate a verification key based on a local variable not predictable in advance of the test. For instance, the current local time may be used, in the form hh:mm:ss, where hh represents hours, mm represents minutes, and ss represents seconds. In this embodiment the local computer uses its local time as input to the PRCG, and generates a pseudo-random code (PRC). The local computer then displays both the local time used as input, and the resulting PRC, which is used as the verification key.

[0085] In this embodiment the proctor may operate the local computer generating the PRC, or, alternatively, the gener-

ation of the PRC may be done at a higher level of the tester organization. In either case, the proctor will have access to both the PRC (the verification key) and the local variable used to generate the key. The proctor may then start the verification application residing on the handheld computer. The verification application requests the local time used by the local-computer PRCG, and then generates the verification key. Since the verification application in the handheld computer contains the same algorithm used by the local-computer PRCG, the handheld computer verification program should produce a verification key equal to that produced by the local computer PRGC.

[0086] Thus, the verification key generated by the handheld computer must be the same as that provided by the tester organization for the proctor to allow the testee to proceed using the flight calculator of the present invention.

[0087] *Fourth Verification Input Embodiment*

[0088] The verification of the contents of the handheld computer in the previous embodiments was accomplished in two steps: first, by authenticating the verifier, presumably the proctor, by the entering of a verification key, and second, having authenticated the verifier, next calculating the checksum of all the applications stored on the handheld

computer, together with the contents of the other memories of the handheld computer, including the RAM, and comparing to the checksum previously stored in the verification program.

[0089] As a further security measure, this embodiment utilizes a PRNG which takes as inputs the verifier's verification key and the checksum, and generates a PRN which is then compared to a result previously stored in the verification program. Thus, in this embodiment, an unscrupulous testee seeking to circumvent the handheld computer's security would have to know the checksum value, the verification key to be input by the proctor, and the algorithm used by the PRNG in order to circumvent the verification security system.

[0090] *Verification Test Result Outputs*

[0091] In its simplest form verification produces a code, such as an alpha-numeric or numeric result, displayed on the handheld computer screen. The proctor has been provided with a code in advance, and if the handheld computer output result does not correspond to this code, the testee will not be allowed to proceed using the flight calculator.

[0092] Alternatively, the proctor may input the expected code

prior to running verification application. The verification results can then be displayed by a simple text output, with the message "SUCCESS" or "FAILURE". However, a testee wishing to circumvent the verification test could conceivable substitute for the verification application an application which accepted the same inputs as the true verification program, but always produced the word "SUCCESS", thus allowing the testee to circumvent the verification, even if illegal data were stored on the handheld computer, in addition to the flight calculator.

[0093] *Encryption Embodiment*

[0094] In order to prevent an unscrupulous testee from disassembling the verification program, the program, in part or in whole, may be encrypted. In the second embodiment, for instance, the table of verification keys alone could be encrypted, since verification keys corresponding to the various test centers might be easily recognized.

[0095] While the invention has been described with reference to specific embodiments, it will be apparent that improvements and modifications may be made within the purview of the invention without departing from the scope of the invention defined in the appended claims.